Università degli Studi di Trieste

Corso di Laurea Magistrale in INGEGNERIA CLINICA

RICHIAMI DI BASI DI DATI

Corso di Informatica Medica

Docente Aleksandar Miladinović

Dipartimento di Ingegneria e Architettura

UNIVERSITÀ DEGLI STUDI DI TRIESTE

DEFINIZIONE E PROPRIETÀ DI UN DATABASE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Un database è un **insieme di dati coerente e logicamente organizzato**, che porta con sé un **significato implicito**:

- **logicamente coerente** (non è una collezione casuale di dati, ma è strutturato in modo da rappresentare informazioni che hanno una relazione fra loro)
- **progettato**, **costruito e popolato** con uno scopo specifico e per utenti definiti
- Rappresenta alcuni aspetti del mondo reale (**minimondo**)

ESEMPIO

UNIVERSITÀ DEGLI STUDI DI TRIESTE

PATIENT	Name	Surname	Address	Location	Unique_ID
	Jack	White	17	MI01	1
	Anna	Green	1	MI03	2
	Herbert	Brown	7	MI01	3

ADDRESS	Street	Number	City	ZIP	State	AddressNumber
	Rockville Pike	3800	New York	19003	NY	1
	Walnut Street	17	Albany	21187	NY	7
	Chestnut Road	17977	Cherry Hill	80087	NJ	17

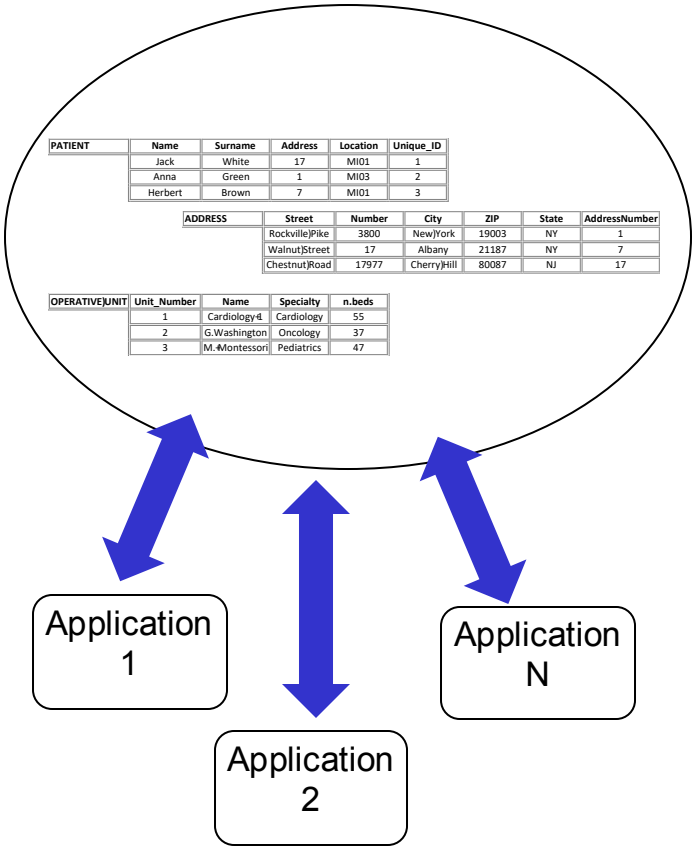
OPERATIVE UNIT	Unit_Number	Name	Specialty	n.beds
	1	Cardiology 1	Cardiology	55
	2	G.Washington	Oncology	37
	3	M. Montessori	Pediatrics	47

DOCTORS	Name	Surname	Specialty	Unique_ID
	Jane	Smith	Surgery	1
	Anne	Powell	Neurology	2
	Henry	Doe	Pharmacology	3

PRESCRIPTIONS	Patient	Operative Unit	Doctor	Drug name
	3	1	3	Paracetamol
	3	2	1	Antibiotics
	1	3	2	Melatonin

DATABASE VS FILE (1)

UNIVERSITÀ DEGLI STUDI DI TRIESTE



Application 1

```
struct Patient {
  char Name[30];
  char Surname[30];
  address Address;
  char location[5];
  char uniqueID [21];
  ....
};
```

Patient FindPatient (char Name[30]);

Application N

```
struct Patient {
  char Name[30];
  char Surname[30];
  address Address;
  char location[5];
  char uniqueID [21];
  ....
};
```

CreateNewPrescription (Patient pat);

DATABASE VS FILE (2)

UNIVERSITÀ DEGLI STUDI DI TRIESTE

DATABASE

- Un unico repository di dati viene mantenuto, definito una volta e poi accessibile da diversi utenti
- Il sistema di database contiene una definizione completa o descrizione del database stesso (natura autosufficiente)
- I programmi di accesso al database sono scritti indipendentemente da qualsiasi file specifico (indipendenza tra programmi e dati)
- Fornisce una rappresentazione concettuale dei dati (astrazione dei dati)
- Ha molti utenti con prospettive diverse (viste multiple)

FILE PROCESSING

- Ogni utente definisce e implementa i file necessari per un'applicazione specifica (ridondanza)
- La definizione dei dati è parte del programma applicativo (definizioni dei dati incorporate)
- La struttura dei file di dati è incorporata nel programma applicativo (dipendenza tra programmi e dati)
- I dati sono rappresentati dall'occupazione della memoria/lunghezza del record
- Ogni utente diverso necessita di un'applicazione diversa (vista unica)

SCHEMI E ISTANZE IN UN DATABASE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

•Lo **SCHEMA DEL DATABASE** è la descrizione del database che viene specificata durante la progettazione del database e non ci si aspetta che cambi frequentemente.

ADDRESS	Street	Number	City	ZIP	State	AddressNumber

OPERATIVE_UNIT	Unit_Number	Name	Specialty	n.beds

•L'**ISTANZA DEL DATABASE** è composta dai dati in un database in un determinato momento (occorrenza o stato). Un nuovo database è un'"istanza vuota."

OPERATIVE_UNIT	Unit_Number	Name	Specialty	n.beds
	1	Cardiology_1	Cardiology	55
	2	G.Washington	Oncology	37
	3	M._Montessori	Pediatrics	47

DATABASE MANAGEMENT SYSTEM
SISTEMA DI GESTIONE DEL DATABASE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Un DBMS è una raccolta di programmi che consente agli utenti di creare e mantenere un database. È un pacchetto software generalizzato per implementare e mantenere un database informatizzato.

Un DBMS svolge tre funzioni principali:

1. **Definizione del database**: Definire un database significa specificare i tipi di dati, le strutture e i vincoli per i dati che saranno memorizzati. Questa fase crea lo scheletro del database, stabilendo cosa può contenere e come deve essere organizzato.
2. **Costruzione del database**: Una volta definito, il database deve essere effettivamente creato e popolato con i dati. Questo processo prevede la memorizzazione dei dati su un supporto fisico, come un disco rigido, gestito dal DBMS.
3. **Manipolazione del database**: Infine, una delle funzioni chiave di un DBMS è permettere la manipolazione dei dati. Questo include operazioni come la ricerca di dati specifici tramite query, l'aggiornamento dei dati per riflettere modifiche nel mini-mondo rappresentato e la generazione di rapporti o report basati sui dati archiviati.

PROPRIETÀ DEL DBMS

UNIVERSITÀ DEGLI STUDI DI TRIESTE

1. Gestione di grandi quantità di dati
2. Fornitura di memoria persistente per oggetti del programma e strutture dati
3. Condivisione dei dati (e controllo della concorrenza)
4. Controllo della ridondanza
5. Fornitura di backup e recupero
6. Limitazione dell'accesso non autorizzato
7. Fornitura di interfacce multiple
8. Rappresentazione di relazioni complesse tra i dati
9. Applicazione dei vincoli di integrità

APPLICAZIONE DEI VINCOLI DI INTEGRITÀ

1. **Coerenza dei dati**: I dati devono essere coerenti, il che significa che lo **stesso tipo di dato** deve essere **sempre rappresentato nello stesso modo** all'interno del database. Ad esempio, se abbiamo un campo per le date, tutte le date devono seguire lo stesso formato e lo stesso tipo di dato.
2. **Relazioni tra i dati**: Le **relazioni tra i dati possono anche essere vincoli importanti**. Questo significa che un record in una tabella deve essere correlato a un altro record in una tabella diversa. Ad esempio, un paziente in un sistema sanitario potrebbe dover essere collegato a un record relativo alla sua cartella clinica.
3. **Valori univoci: Alcuni campi devono contenere valori unic**i, come ad esempio gli ID dei pazienti o i numeri di fattura. Questi vincoli di unicità possono essere verificati direttamente dal sistema, garantendo che non ci siano duplicati che potrebbero creare confusione o errori.
4. **Vincoli semantici**: Infine, i vincoli semantici sono più complessi da verificare automaticamente. Questi si riferiscono alla coerenza logica dei dati rispetto alle regole del mondo reale. Ad esempio, se un sistema deve registrare gli esami universitari, potrebbe essere necessario controllare che i voti siano entro un certo intervallo valido. Tuttavia, alcuni di questi vincoli potrebbero richiedere l'intervento umano per essere interpretati correttamente.

DBMS ARCHITECTURE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Uno dei grandi vantaggi di utilizzare un Database Management System (DBMS) rispetto a un sistema di database non specifico è la sua capacità di offrire una gestione più flessibile e strutturata dei dati

1. **Indipendenza tra dati e applicazioni**: Un DBMS permette una separazione tra i dati e le applicazioni che li utilizzano. Questo significa che possiamo aggiornare o modificare i dati senza dover necessariamente modificare i programmi che li accedono, e viceversa. Questa separazione riduce notevolmente la dipendenza e la complessità nella gestione delle applicazioni.
2. **Viste multiple**: Un altro vantaggio è la possibilità di creare viste multiple dello stesso database. Gli utenti con ruoli diversi possono accedere a differenti versioni o prospettive dei dati, in base alle loro necessità, senza alterare i dati sottostanti.
3. **Uso di un catalogo per lo schema del database**: Il DBMS utilizza un catalogo per memorizzare lo schema del database in modo indipendente dai dati effettivi. Questo schema descrive la struttura del database, quali tabelle contiene, quali campi e vincoli sono presenti, e viene gestito separatamente dai dati stessi. Ciò rende la gestione e la modifica dello schema più facile senza impattare i dati.

I TRE LIVELLI

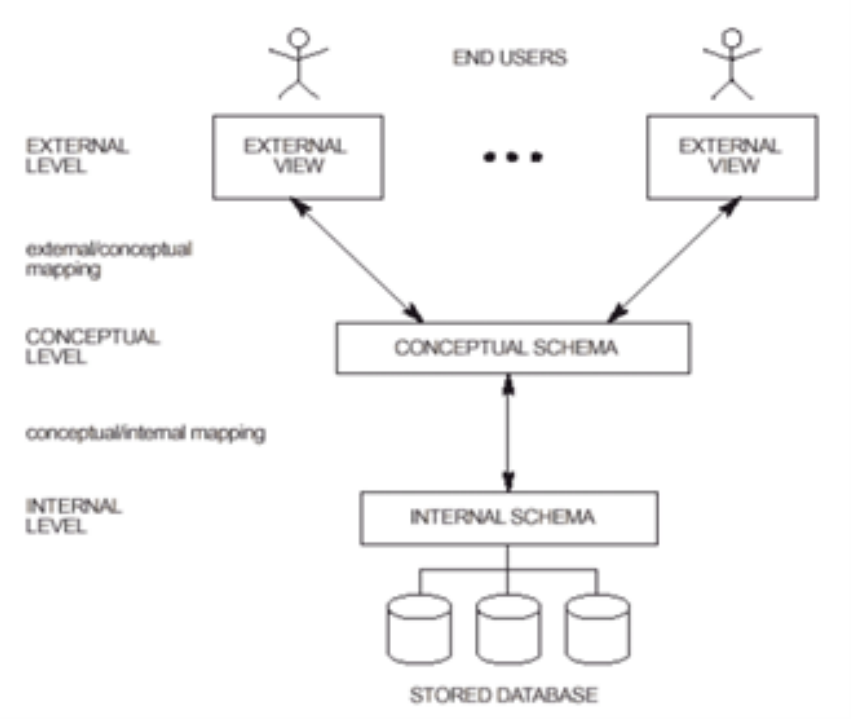
UNIVERSITÀ DEGLI STUDI DI TRIESTE

1. **Livello Interno**: Questo è il livello più basso, che si occupa della struttura fisica di memorizzazione del database. Il **livello interno** contiene lo schema interno, che descrive in dettaglio come i dati sono effettivamente memorizzati, quali sono i percorsi di accesso e le modalità di archiviazione. Questo livello è essenziale per ottimizzare l'accesso e l'archiviazione dei dati, ma è nascosto agli utenti finali.

2. **Livello Concettuale**: A metà strada tra il livello fisico e quello utente, il **livello concettuale** rappresenta la struttura dell'intero database per una comunità di utenti. Qui troviamo lo **schema concettuale**, che descrive la logica del database e le relazioni tra i dati, nascondendo però tutti i dettagli tecnici relativi alla memorizzazione fisica. Questo schema di alto livello viene spesso descritto attraverso modelli concettuali come il modello entità-relazione.

3. **Livello Esterno (Vista)**: Infine, c'è il **livello esterno**, che rappresenta la vista del database per ciascun gruppo di utenti o applicazioni. Ogni **schema esterno** descrive una vista personalizzata del database, mostrando solo le informazioni rilevanti per l'utente specifico e nascondendo tutto ciò che non è di interesse. Questo approccio consente a diversi utenti di accedere agli stessi dati, ma in modalità e prospettive differenti.

I TRE LIVELLI



DEFINIZIONE DI MODELLO DEI DATI

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Un **modello dei dati** è una raccolta di concetti che possono essere usati per descrivere la struttura di un database.

1.Modelli concettuali dei dati → usati per descrivere i dati indipendentemente dal modello logico (es., modello entità-relazione).

2.Modelli rappresentazionali dei dati → usati per rappresentare i dati in un DBMS.

3.Modelli fisici dei dati → descrivono come i dati sono fisicamente memorizzati nella memoria del computer (quanti file, la loro dimensione, ecc.).

MODELLI DI DATI NEI DBMS

UNIVERSITÀ DEGLI STUDI DI TRIESTE

I modelli rappresentazionali dei dati sono usati per fornire una rappresentazione concettuale dei dati in un DBMS.

- Modello di dati gerarchico
- Modello di dati a rete
- Modello di dati relazionale
- Modello di dati orientato agli ogget

CLASSIFICAZIONI DEI DBMS

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Modello dei dati

- **Relazionale**
- A rete
- Gerarchico
- Orientato agli oggetti

Numero di utenti contemporanei

- Utente singolo
- Multiutente

Numero di siti in cui è distribuito il database

- DBMS centralizzato (il database è memorizzato in un unico sito)
- DBMS distribuito (il database è distribuito su molti siti)
- DBMS federato (i database locali hanno un certo grado di autonomia)

MODELLI DI DATI NEI DBMS

I modelli rappresentazionali dei dati sono usati per fornire una rappresentazione concettuale dei dati in un DBMS.

- Modello di dati gerarchico
- Modello di dati a rete
- **Modello di dati relazionale**
- Modello di dati orientato agli ogget

BASIC CONCEPTS

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Il modello relazionale **rappresenta i dati in un database come una raccolta di relazioni.**

Una relazione assomiglia a una tabella → possiamo introdurre una definizione informale.

Tabella = righe, colonne

Relazione = tuple , attributi

RELAZIONI, TUPLE E ATTRIBUTI

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Relational Model

Query Language

Relazione ↔ Tabella

Tupla ↔ Riga

Attributo ↔ Colonna

Relation

Attribute

PATIENT	PHID	FirstName	LastName	Encounter Date	Therapy
	000ZZ000	John	Smith	2003-03-12	Flutamide
	111AA222	Mary	Brown	2004-10-14	Penicillin
	000EE999	Kevin	Green	2001-09-23	Leuprolide
	123XX456	Ann	Black	2002-05-11	Epinephrine

Tuple

RELAZIONI: PROPRIETÀ

UNIVERSITÀ DEGLI STUDI DI TRIESTE

1. Non c'è ordine tra le relazioni (tabelle) in un database relazionale.
2. Non possono esserci due tuple identiche in una relazione (non ridondanza).
3. Gli attributi in una relazione non sono ordinati.
4. Una relazione è caratterizzata da:

   schema della relazione + istanza della relazione.

SCHEMA DI RELAZIONE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Rappresenta "intensione" della relazione:

- Relation Name (eg, PATIENT)
- Relation attributes (eg, PHID, FirstName, LastName, EncounterDate, Therapy)

PATIENT	PHID	FirstName	LastName	Encounter Date	Therapy
	000ZZ000	John	Smith	2003-03-12	Flutamide
	111AA222	Mary	Brown	2004-10-14	Penicillin
	000EE999	Kevin	Green	2001-09-23	Leuprolide
	123XX456	Ann	Black	2002-05-11	Epinephrine

ISTANZA DI RELAZIONE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Rappresenta “estensione” della relazione:

Le tuple (=righe) che contengono i dati reali sono l’istanza della relazione.

PATIENT	PHID	FirstName	LastName	Encounter Date	Therapy
	000ZZ000	John	Smith	2003-03-12	Flutamide
	111AA222	Mary	Brown	2004-10-14	Penicillin
	000EE999	Kevin	Green	2001-09-23	Leuprolide
	123XX456	Ann	Black	2002-05-11	Epinephrine

CHIAVI

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Una Relazione è un insieme di tuple in **cui due tuple non possono essere identiche** (ogni tupla è unica).

Questa proprietà deve essere **valida almeno per un sottoinsieme di attributi.**

Non possono esserci due o più tuple con la stessa combinazione di valori per questo sottoinsieme.

PATIENT	Name	Surname	Address	Location	Unique_ID
	Jack	White	17	MI01	1
	Anna	Green	1	MI03	2
	Herbert	Brown	7	MI01	3
	Jack	White	17	MI02	4

SUPERCHIAVI

UNIVERSITÀ DEGLI STUDI DI TRIESTE

PATIENT	PHID	FirstName	LastName	BirthDate	BirthPlace	GP	Diagnosis
	000ZZ000	John	Smith	1980-03-12	NewYork	Parker	Diabetes
	080JJ333	John	Smith	1945-11-08	Los Angeles	Jackson	Hepatitis
	111AA222	Mary	Brown	1955-10-14	San Antonio	Hart	Hypertension
	000EE999	Kevin	Green	1974-09-23	Sydney	Goldman	Cold
	123XX456	Ann	Black	1963-05-11	Frankfurt	O'Neill	Miocarditis

Superchiave = un sottoinsieme di attributi in una relazione che è unico per ogni tupla.

Una Superchiave identifica univo camente una tupla.

CHIAVI

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Chiave (key) = superchiave (superkey) minima

(una superchiave per cui non è possibile identificare un sottoinsieme di attributi che soddisfi la proprietà di unicità).

- Example:

{FirstName, LastName, BirthDate, BirthPlace, GP} → it is not a key (it is a superkey)

{FirstName, LastName, BirthDate} → it is a key (it is minimal → I cannot exclude any of the attributes, otherwise the tuples are not unique)

{PHID} → superkey and minimal → it is another key of the same relation.

CHIAVI PRIMARIE (PRIMARY KEYS)

UNIVERSITÀ DEGLI STUDI DI TRIESTE

- In una Relazione ci può essere più di una possibile chiave.
- Chiave Primaria = chiave scelta per identificare le tuple in una relazione.
- Notazione: gli attributi che costituiscono la chiave primaria sono seguiti dal simbolo %.

Chiave Primaria - esempi

UNIVERSITÀ DEGLI STUDI DI TRIESTE

1) Chiave Primaria (Primary key) =

{FirstName, LastName, BirthDate}

For the PATIENT relation

PATIENT (PHID, FirstName%, LastName%, BirthDate%, BirthPlace, GP, Diagnosis)

2) Primary key =

{PHID}

For the PATIENT relation

PATIENT (PHID%, FirstName, LastName, BirthDate, BirthPlace, GP, Diagnosis)

VINCOLI DI INTEGRITÀ

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Il concetto di vincoli di integrità deriva dall'osservazione che non tutte le combinazioni di valori sono in grado di rappresentare correttamente le informazioni → l'introduzione di vincoli di integrità garantisce che le informazioni rappresentate siano corrette.

- **Vincoli intra-relazionali**:
  - Vincoli di tupla (NOT NULL, intervallo valido, ecc.)
  - Vincoli di chiave (nessun valore della chiave primaria può essere nullo).

- **Vincoli inter-relazionali**:
  - Vincolo di integrità referenziale (usato per mantenere la coerenza tra le tuple di due relazioni).

VINCOLI DI INTEGRITÀ REFERENZIALE

UNIVERSITÀ DEGLI STUDI DI TRIESTE

- Basati sul concetto di “chiave esterna” (foreign key).

- Un insieme di attributi CHIAVE ESTERNA (FK) nella relazione R1 è una chiave esterna di R1 se:
  - Gli attributi in CHIAVE ESTERNA (FK) hanno lo stesso dominio degli attributi della chiave primaria PK di un'altra relazione R2.
  - Un valore di FK in una tupla t1 di R1 appare come valore di (CHIAVE PRIMARIA) PK in una tupla t2 in R2 o è nullo.

VINCOLO DI INTEGRITÀ REFERENZIALE: ESEMPIO

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Foreign key of PRESCRIPTIONS (FK)

PRESCRIPTIONS	Patient	Operative Unit	Doctor	Drug name
	1	3	1	Paracetamol
	3	2	1	Antibiotics
	1	3	2	Melatonin

1. The two attributes have the same domain
2. The values occurring in Operative Unit occur in Unit_Number and are Primary Keys

OPERATIVE UNIT	Unit_Number	Name	Specialty	n.beds
	1	Cardiology 1	Cardiology	55
	2	G.Washington	Oncology	37
	3	M. Montessori	Pediatrics	47

Primary key of OPERATIVE UNIT (PK)

IL VALORE NULL: SIGNIFICATI MULTIPLI

1. Non valido per l'istanza corrente (es., Cognome del marito per un uomo).
2. Valido ma non ancora esistente (es., Cognome del marito per una donna non sposata).
3. Esistente ma non può essere memorizzato (es., La religione del paziente non può essere memorizzata in alcuni paesi).
4. Esistente ma sconosciuto.
5. Esistente ma non ancora salvato (es., La storia del paziente non ancora raccolta).
6. Memorizzato e poi eliminato (informazioni errate).
7. Disponibile ma in fase di aggiornamento (es., Terapia del paziente in fase di modifica).
8. Disponibile ma non affidabile (una diagnosi non definitiva).
9. Disponibile ma non valida (es., un parametro del sangue al di sopra della soglia).
10. Calcolato da un altro valore NULL (es., BMI se manca il peso).